When Will It Happen? — Relationship Prediction in Heterogeneous Information Networks*

Yizhou Sun[††] Jiawei Han[†] Charu C. Aggarwal[‡] Nitesh V. Chawla[§]
[†] University of Illinois at Urbana-Champaign, Urbana, IL, USA
[‡] IBM T. J. Watson Research Center, Yorktown Heights, NY, USA
[§] University of Notre Dame, Notre Dame, IN, USA
[†]{sun22, hanj}@illinois.edu [‡]charu@us.ibm.com [§]nchawla@nd.edu

ABSTRACT

Link prediction, i.e., predicting links or interactions between objects in a network, is an important task in network analysis. Although the problem has attracted much attention recently, there are several challenges that have not been addressed so far. First, most existing studies focus only on link prediction in homogeneous networks, where all objects and links belong to the same type. However, in the real world, *heterogeneous networks* that consist of multi-typed objects and relationships are ubiquitous. Second, most current studies only concern the problem of *whether* a link will appear in the future but seldom pay attention to the problem of *when* it will happen. In this paper, we address both issues and study the problem of *predicting when a certain relationship will happen in the scenario of heterogeneous networks*. First, we extend the link prediction problem to the relationship prediction problem, by systematically defining both the target relation and the topological features, using a meta path-based approach. Then, we directly model the distribution of relationship building time with the use of the extracted topological features. The experiments on citation relationship prediction between authors on the DBLP network demonstrate the effectiveness of our methodology.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Algorithms

*The work was supported in part by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF-09-2-0053 (NS-CTA), MIAS, a DHS-IDS Center for Multimodal Information Access and Synthesis at UIUC, U.S. National Science Foundation grants IIS-0905215, and U.S. Air Force Office of Scientific Research MURI award FA9550-08-1-0265.
[†]The work was partially performed during Yizhou Sun's employment at IBM T. J. Watson Research Center.

WSDM'12, February 8–12, 2012, Seattle, Washington, USA.

1. INTRODUCTION

Owing to the popularity of Web and social networks, link prediction, *i.e.*, predicting the emergence of links in a network in the future based on certain historical network information, has been a hot topic in recent years. Its applications range from social networks to biological networks, as it addresses the fundamental question of *whether* a link will form between two nodes in the future. Most of the existing link prediction methods [12, 7, 21, 13, 10] are designed for homogeneous networks, in which only one type of objects exists in the network. For example, friendship networks and co-author networks belong to homogeneous networks. However, most of the networks in the real world are heterogeneous, where multiple types of objects and links exist. For example, a movie network contains information about types of movies, actors, users, and comments, with links from types of view/viewed, post/posted, comment-on/commented-on, and so on. Other examples of heterogeneous networks include those extracted from social websites, such as YouTube, Flickr, and Delicious. A few studies [19] have worked on the problem of link prediction in heterogeneous networks, based on the observations of the attributes of the objects. However, the attribute values of objects are usually difficult to fully obtain in reality. For example, user profiles in the movie network is usually incomplete or unreliable.

The heterogeneity of objects and links makes it difficult to use well-known topological concepts in homogeneous networks for algorithmic design. For example, the number of the common neighbors is frequently used as a feature for link prediction in homogeneous networks. However, in heterogeneous networks, the neighbors of an object could come from different types, and the number of shared neighbors is not able to fully represent this heterogeneity. On the other hand, the focus of traditional link prediction tasks is on the fact about **whether** a link will happen in the future, *e.g.*, whether two people will become friends. However, in many applications, it may be more interesting to predict **when** the link will be built. Examples include: "what is the probability that two authors will co-write a paper within 5 years?", and "by when will a user in Netflix rent the movie *Avatar* with 80% probability?".

In this paper, we propose the problem of predicting the *relationship building time* between two objects, based on the topological structure in a heterogeneous network. Different from link prediction which predicts whether a link should exist between two homogeneous typed objects, *relationship prediction* predicts whether or when a relationship between two objects will be built, based on the relationships among

heterogeneous typed objects. We first introduce our framework of relationship prediction in heterogeneous information networks, including the concepts of the target relation and topological features encoded in a meta-path [17]. Then, a generalized linear model (GLM) [5] based supervised framework is proposed to model the relationship building time. In this framework, the building time for relationships are treated as independent random variables with different observations and their expectation is modeled as a function of a linear predictor of the extracted topological features. We propose and compare models with different distribution assumptions for relationship building time, where the parameters for each model are learned separately.

We apply our methodology to predict citation relationship between authors in DBLP, a heterogeneous bibliographic network. The results show that our methodology can indeed handle the task of relationship prediction in heterogeneous networks, and detect the critical topological features in determining the timing of relationship building. By taking the relationship building time into consideration, one can obtain not only the relationship building probabilities within time constrains, but also richer information about the relationship building time, such as median, mean, and quantile intervals. The contributions of this paper are as follows:

- We extend the link prediction problem in homogeneous networks to relationship prediction in heterogeneous networks, by systematically defining the target relation and topological features in heterogeneous networks;
- We extend the traditional prediction problem from "whether it will happen" to "when it will happen", and directly model the relationship building time as a function of topological features; and
- Experiments on citation relationship prediction in the DBLP bibliographic network have validated our methodology.

The remaining of the paper is organized as follows. We introduce the concepts on heterogeneous information networks and define the task of relationship building time prediction in Section 2. Methods for design and calculation of topological features for relationship prediction are developed in Section 3. Relationship building time modeling is worked out in Section 4. We report our experiments in Section 5, discuss related work and other issues in Sections 6, and conclude the study in Section 7.

2. PROBLEM DEFINITION

In this section, we introduce the concepts related to heterogeneous information networks and define the relationship building time prediction task.

2.1 Heterogeneous Information Network

A **heterogeneous information network** is a network containing multiple types of objects and links, which is defined as a directed graph $G = (V, E)$ with a type mapping function $\phi : V \to \mathcal{A}$ and a link mapping function $\psi : E \to \mathcal{R}$. Each object $v \in V$ belongs to one particular type $\phi(v) \in \mathcal{A}$, and each link $e \in E$ belongs to a particular link type or relation $\psi(e) \in \mathcal{R}$. Notice that, if a relation exists between two types A and B, denoted as $A\ R\ B$, the reverse relation R^{-1} holds naturally for $BR^{-1}A$. In most cases, R and its reverse relation R^{-1} are not equal, unless the two types are the same and R is symmetric.

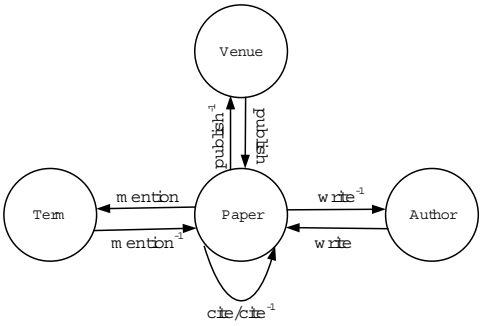

Figure 1: Schema for DBLP Bibliographic Network

Further, in order to describe a heterogeneous network at the meta-level, we use the concept of **network schema** following [17]. Let $G = (V, E)$ be a heterogeneous network with type mapping $\phi : V \to \mathcal{A}$ and relation mapping $\psi : E \to \mathcal{R}$, the network schema of G is a directed graph with nodes from object types $\mathcal{A}$ and edges from relation types $\mathcal{R}$, denoted as $T_G = (\mathcal{A}, \mathcal{R})$.

Here we give one example of heterogeneous networks, namely the DBLP bibliographic network, in its network schema form.

Example 3.1 (DBLP bibliographic network.) The DBLP bibliographic network integrated with citation relationships between papers, which is provided by [18], consists of rich information about publications. The network contains 4 types of objects, namely papers, authors, terms, and venues (conferences or journals). Links exist between authors and papers by the relation of "write" and "written by" (denoted as $write^{-1}$), between papers and terms by "mention" and "mentioned by" (denoted as $mention^{-1}$), between venues and papers by "publish" and "published by" (denoted as $publish^{-1}$), and between papers by "cite" and "cited by" (denoted as $cite^{-1}$). Its network schema is summarized in Fig. 1. ■

For abbreviation, we use the first capital letters to denote these object types, namely P for papers, A for authors, T for terms, and V for venues.

2.2 Target Relation

Given a heterogeneous network, we generalize the link prediction task to **relationship** prediction, which is to predict whether two objects will build a relationship following a certain **target relation** in the future. Notice that target relationships between objects are instances of the target relation. For example, we say that Jim and Mike have built a co-author relationship, if they follow a co-author relation. The target relation can be either a relation in $\mathcal{R}$ or a composite relation concatenated from existing relations. For example, the co-author relation on author set is not defined in our original DBLP network schema, but can be defined through concatenation of two relations "$write$" and "$write^{-1}$", namely two authors a_i and a_j are co-authors, if and only if a_i has authored a paper p that is also authored by a_j.

Formally, we use the concept of **meta-path** [17, 16] defined over the network schema to describe the general relations that can be derived from the network. A meta-path is

a path defined on the graph of network schema $T_G = (\mathcal{A}, \mathcal{R})$, and is denoted in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \ldots \xrightarrow{R_l} A_{l+1}$, which defines a composite relation $R = R_1 \circ R_2 \circ \ldots \circ R_l$ over type A_1 and A_{l+1}, where $\circ$ denotes the composition operator. For example, in the DBLP network, the co-author relation can be described using the meta-path $A \xrightarrow{write} P \xrightarrow{write^{-1}} A$, and is short for $A - P - A$ if there is no ambiguity for either the meaning or order of the relations. Another example is $A - P \rightarrow P - A$, which is short for $A \xrightarrow{write} P \xrightarrow{cite} P \xrightarrow{write^{-1}} A$, and describes the citation relation between authors.

From the topological view, a target relationship is a path instance following the meta-path that defines the target relation. In real cases, the relationship building between two objects can be affected by many factors. In this paper, we are particularly interested in how topological structures in heterogeneous networks affect the relationship building. For example, for the co-authorship prediction, we want to know what kind of connections between two authors in the historical network can help build new co-author relationship in the future. We show that topological features in heterogeneous networks can be defined systematically using meta-paths, which is covered in Section 3.

2.3 Relationship Building Time Prediction

In addition, in this paper we aim at predicting the relationship building time between objects, which differs from the traditional link prediction problem in another aspect. Examples of relationship building time prediction tasks include predicting when two users will be friends in Delicious network; predicting when an author will publish papers in a conference in DBLP network; and predicting when a user will review a movie in IMDB network.

Generally, given a target relation $\langle A, B\rangle$ between two types $A \in \mathcal{A}$ and $B \in \mathcal{A}$, a history interval $T_0 = [t_0, t_1)$, we want to use the topological features extracted from the aggregated network in time period T_0, to predict the relationship building time t ($t \geq t_1$) in the future.

Following the seminal work of link prediction in homogeneous networks [12], we are interested in predicting the generation of new relationships rather than existing ones. Taking the co-authorship prediction as an example, we are interested in predicting the first time of relationship building between two authors who have never co-authored before rather than predicting how many times the existing co-authors will co-author again in the future.

3. TOPOLOGICAL FEATURES IN HETEROGENEOUS NETWORKS

In this section, we study how to systematically define the topological features in heterogeneous networks. Topological features are also called structural features, which are extracted connectivity properties for pairs of objects in the networks. Topological feature-based link prediction aims at inferring the future connectivity by the current connectivity of the network, namely, using the topology of the network itself to infer the evolution of the network.

There are some frequently used topological features defined in homogeneous networks, such as *common neighbors*, *preferential attachment* [2, 14], $katz_\beta$ [9], *Adamic/Adar* [1], and *PropFlow* [13]. Most of these features are either neighbor-based or path-based. Recently, frequent graph patterns are proposed as another topological feature in detecting the link formation rules [4, 11], which can be used for link prediction. However, most of these features are based on homogeneous networks. As there are multi-typed objects and multi-typed relations in heterogeneous networks, the neighbors of an object could belong to multiple types of objects, and the paths between two objects could follow different meta-paths and indicate different relations. For example, for the paths between two authors, it may follow the meta-paths $A - P \rightarrow P - A$, $A - P - V - P - A$, $A - P - T - P - A$ and so on. Thus, we need to design a more sophisticated strategy to generate topological features in heterogeneous networks.

To design the topological features in the heterogeneous networks, we first define the topology using the concept of meta-path, and then define measures to quantify the specific topology.

3.1 Meta Path-Based Topology

As introduced in Section 2, a meta-path is a path defined over a network schema, and denotes a composite relation over a heterogeneous network. Each meta-path defines a unique topology between objects, and can be used to define the topological features with different semantic meanings. In [16], a case study of meta-path preparation for co-authorship prediction is given. In this section, we give a general framework in preparing reasonable meta path-based topological features for the target relation.

In general, for a target relation $R_T = \langle A, B\rangle$, any meta-paths starting with type A and ending with type B other than the target relation itself can be used as the topological features. These meta-paths can be obtained by traversing on the network schema, for example, using BFS (breadth-first search). In particular, we are considering three forms of relations as topological features:

1. $AR_{sim}AR_TB$, where R_{sim} is a similarity relation defined between type A and R_T is the target relation. The intuition is that if a_i in type A is similar to many a_k's in type A that have relationships with b_j in type B, then a_i is likely to build a relationship with b_j in the future.

2. $AR_TBR_{sim}B$, where R_T is the target relation, and R_{sim} is a similarity relation between type B. The intuition is that if a_i in type A has relationships with many b_k's in type B that are similar to b_j in type B, then a_i is likely to build a relationship with b_j in the future.

3. AR_1CR_2B, where R_1 is some relation between A and C and R_2 is some relation between C and B. The intuition is that if a_i in type A has relationships with many c_k's in type C that have relationships with b_j in type B, then a_i is likely to build a relationship with b_j in the future. Notice that the previous two forms are special cases of this one, which can be viewed as triangle connectivity property. Also, $AR_{sim}A$ is another special case of this type.

For topological features, we confine similarity relations R_{sim} and other partial relations R_1 and R_2 to those that can be derived from the network using meta-paths. Moreover, we only consider similarity relations that are symmetric.

Taking the author citation relation, which is defined as $A - P \rightarrow P - A$, as the target relation, we consider 6 author-

Table 1: Meta-Paths Denoting Similarity Relations between Authors

Meta-path	Semantic meaning of the relation
$A-P-A$	a_i and a_j are co-authors
$A-P-A-P-A$	a_i and a_j have the same co-authors
$A-P-V-P-A$	a_i and a_j have publications in the same venues
$A-P-T-P-A$	a_i and a_j write the same terms
$A-P\rightarrow P\leftarrow P-A$	a_i and a_j cite the same papers
$A-P\leftarrow P\rightarrow P-A$	a_i and a_j are cited by the same papers

author similarity relations defined in Table 1. For each similarity relation, we can concatenate the target relation in its left side or in its right side. We then have 12 topology features with the form $AR_{sim}AR_TB$ and $AR_TBR_{sim}B$ in total. Besides, we can consider the concatenation of "author-cites-paper" relation ($A-P\rightarrow P$) and "paper-cites-author" relation ($P\rightarrow P-A$), as well as all the 6 similarity relations listed in Table 1, in the form of AR_1CR_2B. Now we have 19 topological features in total.

For each type of the meta-paths, we illustrate a concrete example to show the possible relationship building in Figure 2. In Figure 2(a), authors a_1 and a_2 are similar, as they publish papers containing similar terms, and a_2 cites papers published by a_3. In the future, a_1 is likely to cite papers published by a_3 as well, since she may follow the behavior of her fellows. In Figure 2(b), author a_1 cites a_2, and a_2 and a_3 are cited by common papers together (p_5, p_6, p_7). Then a_1 is likely to cite a_3 in the future, as she may cite authors similar to a_2. In Figure 2(c), a_1 and a_2 publish in the same venue, then a_1 is likely to cite a_2 in the future as they may share similar interests if publishing in the same conference.

By varying the similarity relations and partial relations, we are able to generate other topological features in arbitrary heterogeneous networks.

3.2 Quantify the Topology Features

Once the topologies defined by meta-paths are determined, the next stage is to propose measures on these meta-paths. We can use the count of the path instances, random walk-based measures, and others to define the measures between any two objects given the meta-path. For similarity relations, say $A-P-T-P-A$, we can treat each term as a 2-hop neighbor of authors and use weighted count or cosine similarity to measure the similarity between two authors. More discussion of measures defined on meta-paths can be found in [17] and [16].

In this paper, without loss of generality, we use the count of path instances as the default measure. Thus, each meta-path corresponds to a measure matrix. For a single relation in $R\in\mathcal{R}$, the measure matrix is just the adjacency matrix of the sub-network extracted by R. Given a composite relation, the measure matrix can be calculated by the matrix multiplication of the partial relations.

In Figure 2(a), the count of path instances between a_1 and a_3 following the given meta-path is 2, which are:

1. $a_1-p_1-t_1-p_2-a_2-p_3\rightarrow p_4-a_3$;

2. $a_1-p_1-t_2-p_2-a_2-p_3\rightarrow p_4-a_3$.

In Figure 2(b), the count of path instances between a_1 and a_4 following the given meta-path is 3, which are:

1. $a_1-p_1\rightarrow p_2-a_2-p_3\leftarrow p_5\rightarrow p_4-a_4$;

2. $a_1-p_1\rightarrow p_2-a_2-p_3\leftarrow p_6\rightarrow p_4-a_4$;

3. $a_1-p_1\rightarrow p_2-a_2-p_2\leftarrow p_7\rightarrow p_4-a_4$.

In Figure 2(c), the count of path instances between a_1 and a_3 following the given meta-path is 1, which is:

1. $a_1-p_1-v_1-p_2-a_3$.

Measures for different meta-paths have different scales. For example, longer meta-paths usually have more path instances due to the adjacency matrix multiplication. We will normalize the measure using Z-score for each meta-path.

3.3 Discussions on Meta Path-based Feature Preparation

How to prepare the meta path-based features is an import issue from a feature engineering point of view. This can be done systematically as well. Note that each meta-path is a path defined on the graph of network schema. For $A\ R\ B$ type of meta-paths, we need to find a path starting from type A and ending with type B. We can use graph traverse algorithms such as BFS (breadth-first search) to enumerate all the possible meta-paths starting from A and ending with B with a length constraint. In particular, if we require $AR_{sim}A$ be a symmetric meta-path between type A to denote a similarity relation, we can confine the meta-path as a round trip path on the network schema. For example, $A-P\leftarrow P\rightarrow P-A$ is a round trip meta-path, which comes back from one type in a reverse manner, while $A-P\rightarrow P-A$ is not a round trip path.

4. MODELING RELATIONSHIP BUILDING TIME

So far, we have provided a systematic way to define the topological features in heterogeneous networks, which is a large space defined over $topology\times measure$.

In this section, we propose the generalized linear model-based prediction model, which directly model the relationship building time as a function of topological features, and provide methods to learn the coefficients of each topological feature, under different assumptions for relationship building time distributions. After that, we introduce how to use the learned model to make inferences.

4.1 Overview

We model the relationship building time prediction problem in a supervised learning framework. In the **training stage**, we first collect the topological features $\mathbf{x}_i$ in the history interval $T_0=[t_0,t_1)$ for each sampled object pair $\langle a_i,b_i\rangle$, where $\phi(a_i)=A$ and $\phi(b_i)=B$. Then, we record their relative first relationship building time $y_i=t_i-t_1$, if t_i is in the future training interval $T_1=[t_1,t_2)$; record the building time $y_i\geq t_2-t_1$, if no new relationship has been observed in T_1. Note that in the training stage, we are only given limited time to observe whether and when two objects will build their relationship, it is very possible that two objects build their relationship after t_2, which needs careful handling in the training model. A generalized linear model (GLM) based relationship building time model is introduced in Section 4, and the goal is to learn the best coefficients associated with each topological feature that maximize the

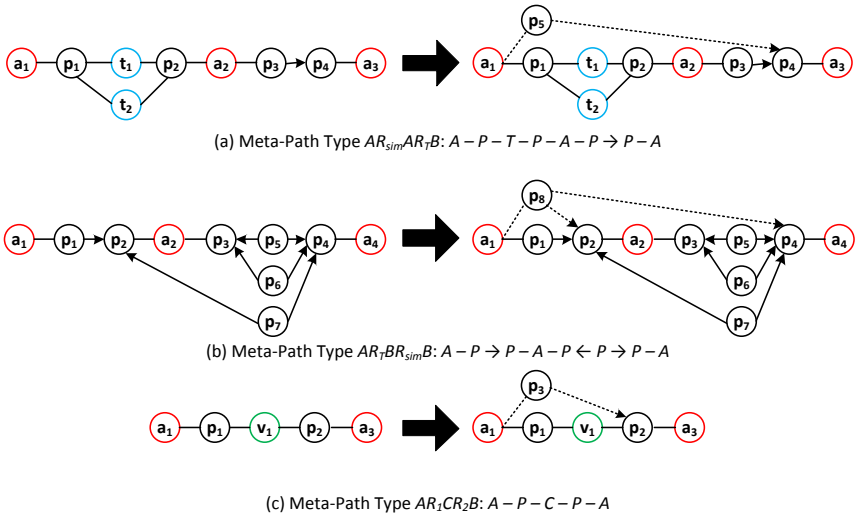

Figure 2: Feature Meta-Path Illustration for Author Citation Relationship Prediction

current observations of the relationship building time. In the **test stage**, we apply the learned coefficients of the topological features to the test pairs, and compare the predicted relationship building time with the ground truth.

Different from the existing link prediction task, in the training stage, we are collecting relationship building time y_i for each training pair, which is a variable ranging from 0 to ∞, rather than a binary value denoting whether a link exists future interval or not. Similarly, in the test stage, we are predicting the relationship building time y_i for test pairs that range from 0 to ∞, rather than predicting whether the link exists or not in the given future interval.

4.2 The Relationship Building Time Prediction Model

We first introduce the generalized linear model [5] as a general framework to solve the relationship building time prediction problem. Then we propose several reasonable distributions for the relationship building time, and use generalized linear model to build the connection between the observed building time and the observed topological features, under different distribution assumptions. We provide the methods to learn the model as well as the model inference in Section 4.3 and 4.4.

4.2.1 The Generalized Linear Model Framework

The main idea of generalized linear model (GLM) [5] is to model the expectation of a dependent random variable Y, $E(Y)$, as some function ("link function") of the linear combination of features, *i.e.*, $\mathbf{X}\beta$, where $\mathbf{X}$ is the observed feature vector, and β is the coefficient vector. Then the goal is to learn β according to the training data set using maximum likelihood estimation. Under different distribution assumptions for Y, usually from the exponential family, $E(Y)$ has different forms of parameter set, and the link functions are with different forms too. Note that the most frequently used Least-Square regression and logistic regression are special cases of GLM, where Y follows Gaussian distribution and Bernoulli distribution respectively.

Suppose we have n training pairs for the target relation $\langle A, B\rangle$. We denote each labeled pair as $r_i = \langle a_i, b_i\rangle$, and y_i as the observed relative relationship building time in the future interval. We denote $\mathbf{X}_i$ as the d dimensional topological feature vector extracted for a_i and b_i in the historical interval plus a constant dimension.

4.2.2 Distributions for Relationship Building Time

The first issue of the prediction model is to select a suitable distribution for the relationship building time. Intuitively, a relationship building between two objects can be treated as an event, and we are interested in when this event will happen.

Let Y be the relationship building time relative to the beginning of the future interval ($y_i = t_i - t_1$), and let T be the length of future training interval. For training pairs, Y has the observations in $[0, T) \cup \{T^+\}$ in a continuous case, and $\{0, 1, 2, \ldots, T-1, T^+\}$ in a discrete case, where $y = T^+$ means no event happens within the future training interval; for testing pairs, Y has the observations in $[0, \infty]$ in a continuous case, and nonnegative integers in a discrete case. In this paper, we consider three types of distributions for relationship building time, namely exponential, Weibull and geometric distribution. For each of the distribution assumptions over y_i, we set up the models separately.

The first distribution is **exponential distribution**, which is the most frequently used distribution in modeling waiting time for an event. The probability density function of an exponential distribution is:

$$f_Y(y) = \frac{1}{\theta}\exp\{-\frac{y}{\theta}\} \tag{1}$$

where $y \geq 0$, and $\theta > 0$ is the parameter denoting the *mean waiting time* for the event. The cumulative distribution function is:

$$F_Y(y) = Pr(Y \leq y) = 1 - \exp\{-\frac{y}{\theta}\} \tag{2}$$

The second distribution is **Weibull distribution**, which is a generalized version of exponential distribution and is another standard way to model the waiting time of an event. The probability density function of a Weibull distribution is:

$$f_Y(y) = \frac{\lambda y^{\lambda-1}}{\theta^\lambda}\exp\{-(\frac{y}{\theta})^\lambda\} \tag{3}$$

where $y \geq 0$, and $\theta > 0$ and $\lambda > 0$ are two parameters related to *mean waiting time* for the event and *hazard of happening* of the event along with the time. λ is also called the shape parameter, as it affects the shape of probability function. When $\lambda > 1$, it indicates an increasing happening rate along the time (if an event does not happen at an early time, it is getting higher probability to happen at later time); and when $\lambda < 1$, it indicates a decreasing happening rate along the time (if an event does not happen at an early time, it is getting less possible in happening in later time). Notice that when $\lambda = 1$, Weibull distribution becomes exponential distribution with mean waiting time as θ, and the happening rate does not change along the time. The cumulative distribution function is:

$$F_Y(y) = Pr(Y \leq y) = 1 - \exp\{-(\frac{y}{\theta})^{\lambda}\} \quad (4)$$

The third distribution is the **geometric distribution**, which is a distribution that models how many times of failures it needs to take before the first-time success. As in our case, the time of failure is the discrete time that we need to wait before a relationship is built. The probability mass function of a geometric distribution is:

$$Pr(Y = k) = (1-p)^k p \quad (5)$$

where $k = 0, 1, 2, \ldots$, and p is the probability of the occurrence of the event at each discrete time. The cumulative distribution function is:

$$Pr(Y \leq k) = 1 - (1-p)^{k+1} \quad (6)$$

In our case, each relationship building is an independent event, and each relationship building time Y_i is an independent random variable, following the same distribution family, but with different parameters. With the distribution assumptions, we build relationship building time prediction models in the following.

4.2.3 Model under Exponential and Weibull Distribution

Notice that, as exponential distribution is a special case of Weibull distribution (with $\lambda = 1$), we only discuss prediction model with Weibull distribution.

In this case, we assume relationship building time Y_i for each training pair is independent of each other, following the same Weibull distribution family with the same shape parameter λ, but with different mean waiting time parameters θ_i. Namely, we assume that different relationships for the target relation share the same trend of hazard happening along with the time, but with different expectation in building time. Under this assumption, the expectation for each random variable Y_i, $E(Y_i) = \theta_i \Gamma(1 + \frac{1}{\lambda})$. We then use the link function $E(Y_i) = \exp\{-\mathbf{X}_i\boldsymbol{\beta}\}\Gamma(1 + \frac{1}{\lambda})$, that is $\log \theta_i = -\beta_0 - \sum_{j=1}^{d} X_{i,j}\beta_j = -\mathbf{X}_i\boldsymbol{\beta}$, where β_0 is the constant term.

Then we can write the log-likelihood function:

$$\log L = \sum_{i=1}^{n} (f_Y(y_i|\theta_i, \lambda) I_{\{y_i<T\}} + P(y_i \geq T|\theta_i, \lambda) I_{\{y_i \geq T\}})$$

where $I_{\{y_i<T\}}$ and $I_{\{y_i \geq T\}}$ are indicator functions, which equals to 1 if the predicate holds, otherwise 0. It is easy to see that the log-likelihood function includes two parts: if y_i is observed in the future interval, we use its real density in the function; otherwise, we are only able to use the probability of $y_i \geq T$ in the function. By plugging in $\log \theta_i = -\mathbf{X}_i\boldsymbol{\beta}$, we can get the log-likelihood with parameters β and λ:

$$LL_W(\boldsymbol{\beta}, \lambda) = \sum_{i=1}^{n} I_{\{y_i<T\}} \log \frac{\lambda y_i^{\lambda-1}}{e^{-\lambda \mathbf{X}_i\beta}} - \sum_{i=1}^{n} (\frac{y_i}{e^{-\mathbf{X}_i\beta}})^{\lambda} \quad (7)$$

We refer this model as Weibull model.

4.2.4 Model under Geometric Distribution

In this case, we assume relationship building time Y_i for each training pair is independent of each other, following the same geometric distribution family, but with different success probability p_i. Under this assumption, the expectation for each random variable Y_i, $E(Y_i) = \frac{1-p_i}{p_i}$. We then let $E(Y_i) = \exp\{-\mathbf{X}_i\boldsymbol{\beta}\}$, namely, $\log \frac{1-p_i}{p_i} = -\mathbf{X}_i\boldsymbol{\beta}$. The log-likelihood function is then:

$$\begin{aligned} & LL_G(\boldsymbol{\beta}) \\ =& \sum_{i=1}^{n} (Pr(Y_i = y_i) I_{\{y_i<T\}} + P(y_i \geq T) I_{\{y_i \geq T\}}) \\ =& \sum_{i=1}^{n} (-I_{\{y_i<T\}}(-\mathbf{X}_i\beta) + (y_i+1)(-\mathbf{X}_i\beta - \log(e^{-\mathbf{X}_i\beta}+1))) \end{aligned} \quad (8)$$

We refer this model as geometric model.

4.3 Model Learning

The learning of the models is becoming an optimization problem, which aims at finding $\hat{\boldsymbol{\beta}}$ and other parameters (*e.g.*, $\hat{\lambda}$ in the Weibull model) that maximize the log-likelihood. As there are no closed form solutions for Eq. 7 and Eq. 8, we use standard Newton-Raphson method to derive the update formulas, which are based on the first derivative and second derivative (Hessian matrix) of the log-likelihood function. The learning algorithms for Weibull model and geometric model are introduced as below.

4.3.1 The Learning Algorithm for Weibull Model

For Weibull model, there are two sets of parameters, namely, the coefficients for each topological feature $\boldsymbol{\beta}$ and the shape parameter λ. An iterative algorithm is proposed to solve this model, where $\boldsymbol{\beta}$ and λ are updated alternatively. Initially, we set $\lambda^{(0)} = 1$, and at the t-th iteration, $\boldsymbol{\beta}$ and λ are updated using the Newton-Raphson method:

- Updating $\boldsymbol{\beta}$ with $\lambda = \lambda^{(t)}$ by setting $\boldsymbol{\beta}^{(t+1)} = \max_{\boldsymbol{\beta}} LL_W(\boldsymbol{\beta}, \lambda^{(t)})$, where $\boldsymbol{\beta}^{(t+1)}$ is derived using an inner iteration of Newton-Raphson method:

  $$\boldsymbol{\beta}^{(t'+1)} = \boldsymbol{\beta}^{(t')} - [H\ LL_W(\boldsymbol{\beta}^{(t')}, \lambda^{(t)})]^{-1} \nabla LL_W(\boldsymbol{\beta}^{(t')}, \lambda^{(t)})$$

  where t' is the inner iteration number, H denotes the Hessian matrix and ∇ denotes the first derivative of function LL_W for $\boldsymbol{\beta}$.

- Updating λ with $\boldsymbol{\beta} = \boldsymbol{\beta}^{(t+1)}$ by setting $\lambda^{(t+1)} = \max_{\lambda} LL_W(\boldsymbol{\beta}^{(t+1)}, \lambda)$, where λ^{t+1} is derived using another inner iteration of Newton-Raphson method:

  $$\lambda^{(t'+1)} = \lambda^{(t')} - [H\ LL_W(\boldsymbol{\beta}^{(t+1)}, \lambda^{(t')})]^{-1} \nabla LL_W(\boldsymbol{\beta}^{(t+1)}, \lambda^{(t')})$$

  where t' is the inner iteration number, H denotes the Hessian matrix and ∇ denotes the first derivative of function LL_W for λ.

Please refer to Appendix A for the concrete formulas for Hessian matrices and first derivatives for $\boldsymbol{\beta}$ and λ. Note that, as for discrete time we may observe 0 for y_i's, which will cause ill-condition of the log-likelihood. Therefore, we will add a small time gap δy to all y_i's in the training period and extract the gap in the test period.

4.3.2 The Learning Algorithm for Geometric Model

For geometric model, there is only one set of parameters $\boldsymbol{\beta}$, and we directly use Newton-Raphson method to optimize the objective function LL_G:

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - [H\ LL_G(\boldsymbol{\beta}^{(t)})]^{-1}\nabla LL_G(\boldsymbol{\beta}^{(t)})$$

where t is the iteration number, H denotes the Hessian matrix and ∇ denotes the first derivative of function LL_G for $\boldsymbol{\beta}$.

Please refer to Appendix B for the concrete formulas for Hessian matrix and first derivative for $\boldsymbol{\beta}$.

4.4 Model Inference

Once the parameters such as $\boldsymbol{\beta}$ and λ are learned from the training data set through MLE, we can apply the model to the test pairs of objects, as long as their topological features in the historical network are given. Let the learned parameter values be $\hat{\boldsymbol{\beta}}$ and $\hat{\lambda}$ for $\boldsymbol{\beta}$ and λ, and let the topological feature vector for the test pairs be $\mathbf{X}_{test}$ (with constant 1 as the first dimension), we now consider three types of questions people may be interested in for the new relationship building time, and provide the solutions in the following.

1. Whether a new relationship between two test objects will be built within t years?

   This question is equal to the query for the probability $Pr(y_{test} \leq t)$, which can be evaluated by plugging in the MLE estimators to derive the distribution parameters. Notice that for traditional link prediction tasks, t should be the same as the length of training interval. For our task, t can be any nonnegative values. For Weibull model, we have:

   $$\begin{aligned} \hat{\theta}_{test} &= \exp\{-\mathbf{X}_{test}\hat{\beta}\} \\ Pr(y_{test} \leq t) &= 1 - \exp\{-(\frac{t}{\hat{\theta}_{test}})^{\hat{\lambda}}\} \end{aligned} \tag{9}$$

   For geometric model, we have:

   $$\begin{aligned} \hat{p}_{test} &= \frac{1}{\exp\{-\mathbf{X}_{test}\hat{\beta}\} + 1} \\ Pr(y_{test} \leq t) &= 1 - (1 - \hat{p}_{test})^{t+1} \end{aligned} \tag{10}$$

2. What is the average relationship building time for two test objects?

   This is simply the query for $E(Y_{test})$. Using the same estimators for $\hat{\theta}_{test}$ and $\hat{p}_{test}$ as above, we can have the estimator for $E(Y_{test})$ as $E(Y_{test}) = \hat{\theta}_{test}\Gamma(1 + \frac{1}{\hat{\lambda}})$ for Weibull model, where $\Gamma(\cdot)$ is the Gamma function, and $E(Y_{test}) = \frac{1-\hat{p}_{test}}{\hat{p}_{test}}$ for geometric model.

3. The quantile: by when a relationship will be built with a probability α?

   This is equal to query for the solution of $F_Y(y_{test}) = \alpha$, and we can get answers as $y_{test} = \hat{\theta}_{test}(-\log(1-\alpha))^{\frac{1}{\hat{\lambda}}}$ for Weibull model, and $y_{test} = \max\{\frac{\log(1-\alpha)}{\log(1-\hat{p}_{test})} - 1, 0\}$ for geometric model. When $\alpha = 0.5$, the quantile is just the median.

5. EXPERIMENTS

In this section, we apply our methodology on DBLP bibliographic network, and select the target relation as the author citation relationship ($A - P \rightarrow P - A$). The goal is to study the effectiveness of our time-involved relationship prediction model in the heterogeneous network scenario.

5.1 The Dataset

We select a subset of authors in the DBLP bibliographic network, who published more than 5 papers in top conferences in four areas[1] that are related to data mining between year 1996 and 2000 ($T_0 = [1996, 2000]$). The total number of the author set is 2721. Then we sampled 7000 pairs of authors in the form of $\langle a_i, a_j \rangle$ that a_i did not cite a_j in T_0, but have citation relationship between year 2001 and 2009 ($T_1 = [2001, 2009]$ and $T = 9$) as positive samples; and we sampled another 7000 pairs of authors that have no citation relationship during either T_0 or T_1. The citation relationship is defined if a_i cites papers written by a_j published before year 2000. Notice that, we have this time constraint for papers as we want to infer citation relationship via the historical network. 19 topological features introduced in Sec. 3 are calculated for each year in T_0 and then aggregated together. The first (relative) time of the citation relationship is recorded for each pair of authors; and if there is no citation relationship between them in T_1, the time is recorded as a value bigger than 9.

5.2 Experimental Setting

In order to show the power of using time-involved model in relationship prediction, we use logistic regression [15] (denoted as *logistic*) that is frequently used in binary link prediction tasks as the baseline. Notice that, the output of the logistic regression is a probability denoting whether a relationship will be built in T_1 for each test pair. In our models, the output is the parameter set for the distribution of the relationship building time, from which we can infer much more information rather than a simple probability. We denote our models with different distribution assumptions as *GLM_geo*, *GLM_exp*, and *GLM_weib* respectively.

To compare the four models, we use two sets of measures to evaluate the effectiveness of each model. First, we measure the effectiveness according to the predicted probability for each relationship. We define the *accuracy* of the relationship prediction as the ratio between the number of correctly predicted relationship (under the *cut-off* 0.5) and the total number of the test pairs. Also, another frequently used measure AUC (the area under ROC curve) [3] is used to compare the accuracy.

Second, we directly compare the predicted time with the ground truth, among our proposed models. Mean absolute error (*MAE*) that is the mean of the absolute error between predicted relationship building time and the ground truth is used. Also, we use the ratio of the relationships that occur in some confidence interval derived from the models as another measure to test the accuracy of the predicted time. Notice that, relationships yet to happen are not considered in these two measures.

[1]Data Mining: KDD, PKDD, ICDM, SDM, PAKDD; Database: SIGMOD Conference, VLDB, ICDE, PODS, EDBT; Information Retrieval: SIGIR, ECIR, ACL, WWW, CIKM; and Machine Learning: NIPS, ICML, ECML, AAAI, IJCAI.

Table 2: Relationship Prediction Accuracy Comparison

	$T=1$		$T=5$		$T=9$	
	Acc.	AUC	Acc.	AUC	Acc.	AUC
logistic	**0.9312**	**0.7356**	**0.7097**	0.7751	0.6995	**0.8083**
GLM-geo	0.9310	0.7037	0.6909	**0.7758**	0.6659	0.8021
GLM-exp	0.9304	0.7262	0.6922	0.7680	**0.7096**	0.7917
GLM-weib	0.9304	0.7273	0.6915	0.7680	0.7031	0.7917

All the results in this section are the average results using 10-fold cross-validation.

5.3 Prediction Power Study

We now compare our time-involved models with the baseline logistic regression, using the first set of measures.

First, we vary the future interval used in the training stage T_1^{train} (with the length T^{train}) and test stage T_1^{test} (with the length T^{test}) but force them the same ($T = T^{train} = T^{test}$), and compare the predicted probability of the relationship building within time T ($Pr(y_{test}) < T$) for test pairs. The results are summarized in Table 2. From the results, we can see that logistic regression has the best overall performance in predicting the *probability* of relationship building, when the training future interval equals to the test future interval.

Next, we test the generality power for different models, namely, when the training future interval is not equal to the test future interval ($T^{train} \neq T^{test}$). On one hand, we may want to know the probability of relationship building within each year in the training interval ($T^{test} < T^{train}$); on the other hand, we may want to infer longer term probability given a short term training interval ($T^{test} > T^{train}$). We show the two cases in Table 3 and Table 4. Notice that, since logistic regression can only output the probability when $T^{test} = T^{train}$, we use the same predicted probability for different test intervals. In Table 3, we fix the training interval with length $T^{train} = 9$, namely, $T_1^{train} = [2001, 2009]$, and vary the test intervals with length from 1 to 4. We can see that when T^{test} is small, time-involved models can give much better prediction accuracy, especially in terms of the measure *accuracy*. In other words, time-involved models carry more information in telling the probability of relationship building in finer time periods. In Table 4, we fix the test interval with length $T^{test} = 9$ and vary the training intervals with length from 2 to 5. We can see that, time-involved models can better utilize the short term training than logistic regression, and output better prediction results for longer term relationship building behavior. It is interesting to notice that by using the measure AUC, which does not require users to specify a *cut-off* value in the predicted probabilities, the performance of logistic regression is still comparable with other models. This is due to AUC only uses the ranking order of the predicted values, while *accuracy* requires that the *absolute values* of the predicted probabilities are also correct.

In all, for time-involved model, it contains more information and can answer different questions and with strong generalization power. Logistic regression can only answer the question of whether a link will happen or not, given a fixed time interval, and the experiments show that it has the strength in answering this certain type of question. However, if we are asking more, it fails in most of the scenarios.

5.4 Time Prediction Accuracy Study

We now evaluate the predicted time using different time-involved models. Here, we use the predicted median time as the predicted time. Table 5 shows the MAE between the predicted median time and the ground truth under different training and test intervals. It turns out that *GLM-exp* has the lowest error. Also, both *GLM-exp* and *GLM-weib* perform even better using shorter interval as training, whereas *GLM-geo* has the opposite behavior, that is, longer term of training leads to better performance. Notice that, we only calculate the error for the positive relationships happened in the test interval.

Table 5: *MAE* of Predicted Time with the Ground Truth

	$T^{train}=5, T^{test}=9$	$T^{train}=9, T^{test}=9$
GLM-geo	4.9883	4.7219
GLM-exp	**2.7774**	**3.0685**
GLM-weib	3.1025	3.1692

In Table 6, we infer different confidence intervals from the predicted relationship building time distribution, and test the ratio of the true relationship in different confidence intervals. A confidence interval (range) rather than a simple value, say the median time, can give users a better view of the relationship building time. It is shown that *GLM-exp* and *GLM-weib* has a higher ratio of giving correct confidence intervals for the true relationship building time, especially when using a small confidence interval. This is very useful in practice as they can give tight bound estimations.

5.5 Case Studies

To better understand the output of our model, we now show a case study of citation relationship between "Philip S. Yu" and other candidates. The model is trained by *GLM-weib* using a training interval of 9 years ($T_1^{train} = [2001, 2009]$), with the learned parameter $\lambda = 0.9331$, slightly less than 1, which means the citation relationship has a higher hazard happening at an earlier time. The ground truth of the citation building time, and the predicted median, mean, 25% quantile and 75% quantile for several test pairs are shown in Table 7. It can be seen that the predicted median and confidence interval are very suggestive for predicting the true citation relationship building time. For those authors whose predicted being cited time is significantly different from the ground truth, in-depth studies may be needed. For example David Maier is a prolific researcher in database area, and by intuition as well as suggested by the model, Philip should cite him. However, the ground truth says otherwise. Furthermore, this function can be used to recommend authors to any author in DBLP for citation purpose.

For the above model, the learned *top-4* most important topological features with the highest coefficients are:

1. $A - P - T - P - A$, namely, if two authors are very similar in terms of writing similar topics, they tend to cite each other;
2. $A - P \leftarrow P \rightarrow P - A$, namely, if two authors are very similar in terms of being frequently co-cited by the common papers, they tend to cite each other;
3. $A - P - A - P \rightarrow P - A$, namely, an author tends to cite the authors that are frequently cited by her co-authors;
4. $A - P - T - P - A - P \rightarrow P - A$, namely, if two authors are similar in terms writing similar topics, they tend to cite the same authors.

These topological features provide insightful knowledge for people in understanding the citation relationship building.

Table 3: Prediction Generalization Power Comparison: $T^{test} < T^{train}$ and $T^{train} = 9$

	$T^{test} = 1$		$T^{test} = 2$		$T^{test} = 3$		$T^{test} = 4$	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
logistic	0.7106	0.7619	0.7246	**0.7535**	0.7669	0.7347	**0.7349**	**0.7731**
GLM-geo	0.9284	**0.7626**	0.8436	0.7532	**0.7829**	**0.7657**	0.7347	0.7696
GLM-exp	**0.9290**	0.7553	**0.8442**	0.7464	0.7821	0.7569	0.7328	0.7603
GLM-weib	0.9287	0.7273	0.8441	0.7452	0.7826	0.7559	0.7334	0.7597

Table 4: Prediction Generalization Power Comparison: $T^{test} > T^{train}$ and $T^{test} = 9$

	$T^{train} = 2$		$T^{train} = 3$		$T^{train} = 4$		$T^{train} = 5$	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
logistic	0.5157	0.7810	0.5379	0.7805	0.5599	0.7841	0.5952	0.7896
GLM-geo	0.5942	**0.7910**	0.6209	**0.7926**	0.6366	**0.7902**	0.6522	**0.7982**
GLM-exp	0.5015	0.7802	0.5214	0.7833	0.6709	0.7841	**0.7143**	0.7870
GLM-weib	**0.7081**	0.7816	**0.7021**	0.7832	**0.7002**	0.7833	0.7103	0.7862

6. RELATED WORK

The link prediction problem has been first studied on homogeneous networks. Early works mainly study unsupervised methods [1, 12], namely they propose different similarity measures according to either topological structures of the networks or proximity of object attributes that are consistent with the link appearance in the future. Later, supervised methods that are able to combine different features with different coefficients via training data sets are proposed by different studies [7, 21, 13]. A recent study [10] has discussed the link prediction problem when the network is not fully observed and thus is modeled in a probabilistic way. A survey in link prediction can be found in [6]. In this paper, we extend the link prediction problem to the more general heterogeneous networks, by extending link prediction to relationship prediction and exploring the topological features in such scenarios.

Recently, some studies [4, 11] propose frequent graph pattern mining-based methodology to detect graph evolution rules, which provides some clues for proposing new topological features in the network for link prediction. However, the focus on the two papers are still on homogeneous networks, and they have not considered how different frequent evolution patterns affect the link formation speed yet. That is, this methodology cannot answer the "when" problem of link formation.

Another line of study similar to our problem is the link prediction task in relational data [15, 19], as relational data also involves different types of objects and complex relationships between objects. However, these studies have a focus different from our paper. As in [15], they study feature selection in a relational environment using relational languages, and feed these features into supervised link prediction models; for [19], their goal is to model the relational data via a probabilistic model. In our paper, we aim at designing a model for relationship building time by systematically exploring the topological features in heterogeneous networks.

A novel meta path-based similarity measure called PathSim is proposed in [17], which also defines the framework of similarity measure in a heterogeneous network scenario. In [16], the authors study the relationship prediction problem using co-authorship prediction as a case study in the heterogeneous network. However, they have not systematically study the relationship prediction in a general case and they are only focus on the "whether" problem rather than the "when" problem. In this paper, we build a framework for general relationship prediction in heterogeneous networks by systematically extracting meta path-based topological features, and study when the relationship will happen in the future.

The general setting of link prediction task is set by Liben-Nowell and Kleinberg [12], which is to predict *whether* a link between two existing objects will be added to the network during the time interval $[t, t + \Delta t]$ given the snapshot of the network at time t. In other words, the task has not considered the issue *when* a link will appear in this time interval. Recently, several studies have considered the extension on usage of time. In [20], a methodology that assigns weights to events and edges according to their appearing time is proposed, which produces better link prediction accuracy by using more time information in the feature side. In [8], a time series model is proposed to predict the frequency of repeated links in networks. In comparison to these studies, our paper focuses on the new relationship prediction and aims at modeling the relationship building time in the future.

In all, in this paper, we extend the traditional link prediction in homogeneous networks into relationship prediction in the more complex heterogeneous networks, and aims at modeling the relationship building time in the future.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a new problem of study relationship building time prediction in heterogeneous networks. This problem on one hand extends the traditional link prediction problem in homogeneous networks into relationship prediction in heterogonous networks; and on the other hand it extends the traditional "whether a link will happen in a fixed future interval" to "when it will happen." We systematically study how to define relationship and how to select topological features in heterogeneous networks, using a meta path-based concept. Time-involved relationship prediction models are proposed. Experiments have shown the effectiveness of our proposed models.

Predicting relationship building time is still an open problem. It is worth of considering a broader range of target relations in different heterogeneous networks, and propose a systematic methodology of model selection and model comparison. More future work can be done along this line.

8. REFERENCES

[1] L. A. Adamic and E. Adar. Friends and neighbors on the web. *SOCIAL NETWORKS*, 25:211–230, 2001.

Table 6: Ratio of the True Relationship Occurring in Different Confidence Intervals: $T^{test} = 9$

	25%-75%		10%-90%		0%-80%	
	$T^{train} = 9$	$T^{train} = 5$	$T^{train} = 9$	$T^{train} = 5$	$T^{train} = 9$	$T^{train} = 5$
GLM-geo	0.5489	0.5336	**0.8936**	**0.8947**	0.9650	0.9743
GLM-exp	0.7167	0.7246	0.8619	0.8634	0.9880	0.9889
GLM-weib	**0.7278**	**0.7314**	0.8680	0.8686	**0.9884**	**0.9896**

Table 7: Case Studies of Relationship Building Time Prediction

a_i	a_j	Ground Truth	Median	Mean	25% quantile	75% quantile
Philip S. Yu	Ling Liu	1	2.2386	3.4511	0.8549	4.7370
Philip S. Yu	Christian S. Jensen	3	2.7840	4.2919	1.0757	5.8911
Philip S. Yu	C. Lee Giles	0	8.3985	12.9474	3.2450	17.7717
Philip S. Yu	Stefano Ceri	0	0.5729	0.8833	0.2214	1.2124
Philip S. Yu	David Maier	9+	2.5675	3.9581	0.9920	5.4329
Philip S. Yu	Tong Zhang	9+	9.5371	14.7028	3.6849	20.1811
Philip S. Yu	Rudi Studer	9+	9.7752	15.0698	3.7769	20.6849

[2] A. L. Barabasi, H. Jeong, Z. Neda, E. Ravasz, A. Schubert, and T. Vicsek. Evolution of the social network of scientific collaborations. *arXiv:cond-mat/0104162v1*, Apr 2001.

[3] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, July 1997.

[4] B. Bringmann, M. Berlingerio, F. Bonchi, and A. Gionis. Learning and predicting the evolution of social networks. *IEEE Intelligent Systems*, 25:26–35, 2010.

[5] A. J. Dobson. *An Introduction to Generalized Linear Models, Second Edition*. Chapman & Hall/CRC, November 2001.

[6] L. Getoor and C. P. Diehl. Link mining: a survey. *SIGKDD Explor. Newsl.*, 7:3–12, December 2005.

[7] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *Proc. of SDM '06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[8] Z. Huang and D. K. J. Lin. The time-series link prediction problem with applications in communication surveillance. *INFORMS J. on Computing*, 21:286–303, April 2009.

[9] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39 – 43, 1953.

[10] V. Leroy, B. B. Cambazoglu, and F. Bonchi. Cold start link prediction. In *KDD '10*, 2010.

[11] C. W.-k. Leung, E.-P. Lim, D. Lo, and J. Weng. Mining interesting link formation rules in social networks. In *CIKM '10*, 2010.

[12] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM '03*, pages 556–559, 2003.

[13] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *KDD '10*, 2010.

[14] M. E. J. Newman. Clustering and preferential attachment in growing networks. *Physical Review Letters E*, 64, 2001.

[15] A. Popescul, R. Popescul, and L. H. Ungar. Statistical relational learning for link prediction. In *Proc. of the Workshop on Learning Statistical Models from Relational Data at IJCAI-2003.*, 2003.

[16] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han. Co-author relationship prediction in heterogeneous bibliographic networks. In *ASONAM' 11*, 2011.

[17] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. In *VLDB' 11*, 2011.

[18] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: extraction and mining of academic social networks. In *KDD '08*, 2008.

[19] B. Taskar, M. fai Wong, P. Abbeel, and D. Koller. Link prediction in relational data. In *NIP '03*, 2003.

[20] T. Tylenda, R. Angelova, and S. Bedathur. Towards time-aware link prediction in evolving social networks. In *Proceedings of the 3rd Workshop on Social Network Mining and Analysis*, SNA-KDD '09, pages 9:1–9:10, 2009.

[21] C. Wang, V. Satuluri, and S. Parthasarathy. Local probabilistic models for link prediction. In *ICDM '07*, pages 322–331, 2007.

APPENDIX

A. FORMULAS FOR WEIBULL MODEL

- First derivative and Hessian matrix for $\boldsymbol{\beta}$:

$$\nabla LL_W(\boldsymbol{\beta}) = -(-\lambda \mathbf{X}'(I_{\{Y<T\}}) + \lambda \mathbf{X}' \exp\{\lambda(\log(Y)\mathbf{X}\boldsymbol{\beta})\})$$
$$H\ LL_W(\boldsymbol{\beta}) = -\lambda^2((\mathbf{X}. * (\exp\{\lambda(\log(Y) + \mathbf{X}\boldsymbol{\beta})\}\mathbf{1}_{1\times p}))'\mathbf{X})$$

- First derivative and second derivative for λ:

$$\nabla LL_W(\lambda) = \sum_i (\frac{I_{\{y_i<T\}}}{\lambda} + \lambda(\log(y_i) + \mathbf{X}_i\boldsymbol{\beta}) - (\log(y_i) + \mathbf{X}\boldsymbol{\beta})\exp\{\lambda(\log(y_i) + \mathbf{X}_i\boldsymbol{\beta})\})$$
$$H\ LL_W(\lambda) = \sum_i (-\frac{I_{\{y_i<T\}}}{\lambda^2} - (\log(y_i) + \mathbf{X}_i\boldsymbol{\beta})^2\exp\{\lambda(\log(y_i) + \mathbf{X}_i\boldsymbol{\beta})\})$$

B. FORMULAS FOR GEOMETRIC MODEL

- First derivative and Hessian matrix for $\boldsymbol{\beta}$:

$$\eta = \exp\{-\mathbf{X}\boldsymbol{\beta}\}$$
$$\nabla LL_G(\boldsymbol{\beta}) = -\mathbf{X}'(-(I_{\{Y<T\}}) + Y + \mathbf{1}_{n\times 1})./(\eta + \mathbf{1}_{n\times 1})$$
$$H\ LL_G(\boldsymbol{\beta}) = -((Y + \mathbf{1}_{n\times 1}). * \eta./(\eta + \mathbf{1}_{n\times 1}).^2))\mathbf{X}\mathbf{1}_{1\times p})'\mathbf{X}$$